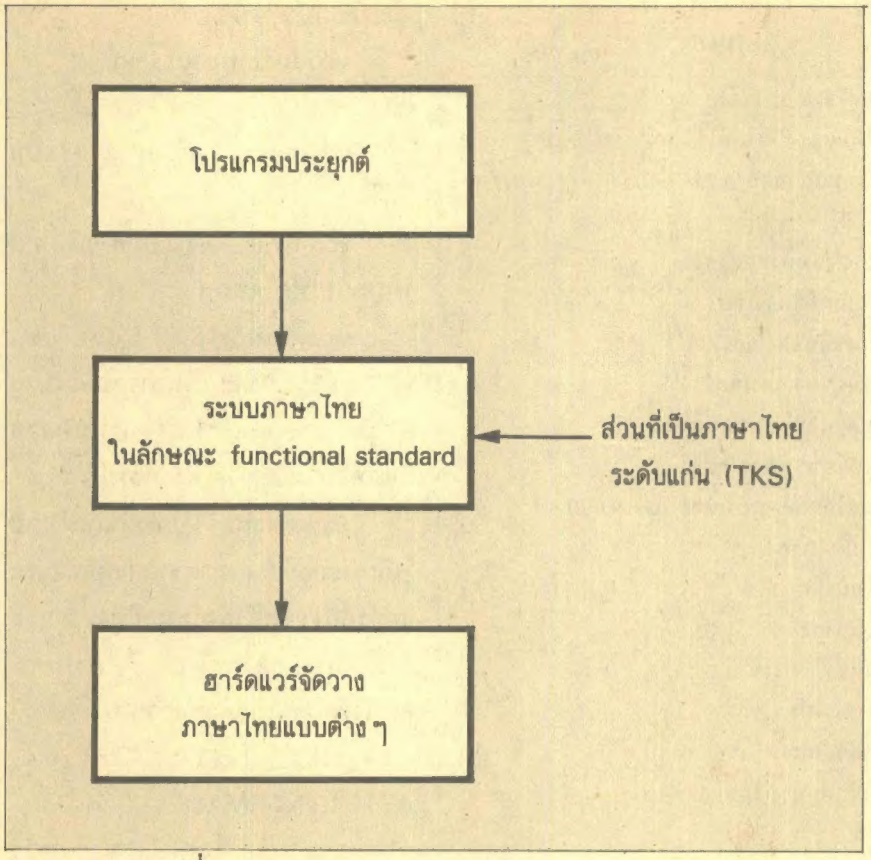
ยืน ภู่วรวรรณ
จเร เลิศสุดวิชัย
พรพจน์ สุริยะวงศ์

บทนำ

ในการพัฒนาซอฟต์แวร์ที่ใช้กับระบบคอมพิวเตอร์ ผู้ออกแบบจะต้องวางโครงร่างของระบบซอฟต์แวร์ให้ดีตั้งแต่เริ่มต้น มีการพัฒนาเครื่องมือที่จะนำมาใช้ต่อไป การพัฒนาภาษาไทยระดับแก่นนี้มีจุดมุ่งหมายเพื่อให้เห็นแนวทางของการกำหนด **มาตรฐานระดับฟังก์ชัน (Functional Standard)** ขึ้นในประเทศไทย โดยมาตรฐานนี้จะกำหนดในลักษณะฟังก์ชันว่าควรจะมีฟังก์ชันการประมวลผลภาษาไทยอะไรบ้าง แต่ละฟังก์ชันมีหน้าที่อย่างไร การประมวลผลกระทำในลักษณะใดเพื่อให้การเรียกใช้ในระดับสูงขึ้นทำได้ดีขึ้น

บทความชุดนี้จึงเป็นบทความแนวเสนอแนะเพื่อให้เห็นหลักการหรือให้เห็นความสำคัญของระดับ functional standard เพื่อให้ระบบภาษาไทยที่มีการพัฒนากันขึ้นมีมาตรฐานในระดับผู้ใช้ โดยโครงสร้างของระบบจะเป็นดังรูปที่ 1

การที่เราสามารถจัดการในลักษณะ functional standard ได้จะทำให้การพัฒนาซอฟต์แวร์ของเราในระดับประยุกต์เป็นไปได้ดีขึ้น และสามารถใช้กับฮาร์ดแวร์จัดการภาษาไทยในหลายรูปแบบ เพื่อให้เกิดการใช้งานได้กว้างขวาง



รูปที่ 1 การวางมาตรฐาน functional standard

ยิ่งขึ้น สำหรับในตอนนี้จะได้ให้รายละเอียดของการแสดงผลบนจอภาพในระดับ TKS ก่อน

ใครเป็นผู้พัฒนาระบบภาษาไทยระดับแก่น

เนื่องจากตัวอย่างและระบบที่เสนอนี้เป็นเพียงรูปแบบที่พัฒนาขึ้นที่ห้องปฏิบัติการที่กลุ่มผู้เขียนทำงานอยู่ โดยหลักการแล้วสถาบันมาตรฐานของประเทศจะเป็นผู้กำหนด functional standard เพื่อการเชื่อมต่อนี้ขึ้นมาก่อน จากนั้นกลุ่มพัฒนาระบบภาษาไทยหรือผู้ทำไทยการ์ดทุกบริษัทก็จะพัฒนาซอฟต์แวร์ไดรฟ์เวอร์ระดับแก่นนี้ การพัฒนากันขึ้นแต่ละกลุ่มจะมีโครงสร้างภายในที่แตกต่างได้ แต่ระดับการเชื่อมต่อหรือการเรียกเข้าหาฟังก์ชันต้องเป็นมาตรฐานเดียวกัน

สำหรับตัวอย่างของการพัฒนาในระบบ TKS ในบทความนี้ เป็นการพัฒนาในระบบที่ใช้กับฮาร์ดแวร์เดิมของเครื่องไมโครคอมพิวเตอร์ 16 บิต ซึ่งยึดหลักการแสดงผลแบบบิตกราฟิก คือ การแสดงภาษาไทยจะไม่จำเป็นต้องแก้ไขหรือปรับปรุงฮาร์ดแวร์ใด ๆ เลย ขอบเขตของการแสดงผลจะเป็นไปตามมาตรฐานเดิม คือ จอภาพแบบ CGA จะแสดงที่ความละเอียด 640×200 จุด และแบบเฮอร์คิวลิสจะแสดงผลด้วยความละเอียด 720×350 จุด สำหรับจอภาพที่มีความละเอียดสูงกว่านี้ก็จะแสดงผลได้เช่นกัน

ดังนั้นหากผู้พัฒนาไทยการ์ดอื่น ๆ ที่พัฒนาโปรแกรมให้ได้ฟังก์ชันตามข้อกำหนดนี้ ก็จะทำให้โปรแกรมประยุกต์อื่นพัฒนามาตามเงื่อนไขของ TKS นี้ทำงานได้

TKS ในระดับ PDD

จากที่กล่าวไว้แล้วในตอนก่อนว่า โครงสร้างของ TKS มีสามส่วนคือ ส่วนของ PDD หรือ Physical Device Driver เป็นส่วนที่ใกล้ชิดกับฮาร์ดแวร์มากที่สุด กับส่วนที่สองคือ LDD หรือ Logical Device Driver เป็นส่วนที่ใช้สำหรับการเปลี่ยนทิศทางของการเรียกเข้าหาระบบ LDD จะมีประโยชน์มากสำหรับการทำงานในระดับโอเอส เช่นการนำ TKS ไปใช้ใน OS/2 หรือ UNIX ส่วนนี้จะทำให้การปรับเข้าสู่ระบบทำได้ง่ายขึ้น ส่วนของ TKS ส่วนสุดท้ายคือ FCLF-Function Call and Library Function เป็นส่วนของไลบารี่ที่เก็บรวบรวมฟังก์ชันที่เป็นประโยชน์อื่น ๆ ไว้

PDD ในส่วนแสดงผลบนจอภาพ

การแสดงผลบนจอภาพในส่วน TKS ที่เสนอนี้กำหนดให้แสดงผลบนจอภาพเป็นแบบกราฟิก โดยให้โปรแกรมประยุกต์ต่าง ๆ มองจอภาพเป็นเท็กซ์ ส่วนของ TKS จะทำการแสดงผลให้เสมือนเป็นโหมดตัวอักษร

การกำหนดฟังก์ชันการเชื่อมต่อกับจอภาพในระดับ PDD กำหนดเป็นหมายเลขโดยมีรายละเอียดแสดงได้ดังตารางที่ 1

การเรียกใช้ฟังก์ชันในระดับ PDD

ฟังก์ชันของ PDD เป็นโปรแกรมที่เขียนขึ้นเพื่อจัดการกับการแสดงผลบนจอภาพโดยตรง สำหรับการประยุกต์

หมายเลข	ชื่อฟังก์ชัน	ความหมาย
0	setup video system	กำหนดโหมดของวิดีโอ
1	get current video system	อ่านโหมดของวิดีโอที่ใช้
2	set video system ON/OFF	เซตระบบแสดงผล (ON/OFF)
3	reserve	
4	set cursor position	กำหนดตำแหน่งเคอร์เซอร์
5	get cursor position	อ่านตำแหน่งเคอร์เซอร์
6	set cursor type	กำหนดชนิดของเคอร์เซอร์
7	get cursor type	อ่านชนิดของเคอร์เซอร์
8	set cursor style	กำหนดรูปแบบของเคอร์เซอร์
9	get cursor style	อ่านรูปแบบของเคอร์เซอร์
10	set cursor on/off	กำหนดให้แสดงเคอร์เซอร์ (ON/OFF)
11	move screen block	เคลื่อนย้ายภาพ
12	scroll up	เลื่อนหน้าขึ้น
13	scroll down	เลื่อนหน้าลง
14	scroll left	เลื่อนหน้าไปทางซ้าย
15	scroll right	เลื่อนหน้าไปทางขวา
16-19	display character with attribute	แสดงตัวอักษรพร้อมแอตทริบิวต์
20-23	display character with current attribute	แสดงตัวอักษรพร้อมแอตทริบิวต์
24-25	display structure character and attribute	แสดงอักษรเป็นโครงสร้างพร้อมแอตทริบิวต์
26	read character with attribute	อ่านตัวอักษรพร้อมแอตทริบิวต์
27	read structure character with attribute	อ่านอักษรเป็นโครงสร้างพร้อมแอตทริบิวต์
32	display stream with attribute	แสดงสตรีมของอักษรพร้อมแอตทริบิวต์
33	display text with attribute	แสดงเท็กซ์ของตัวอักษรพร้อมแอตทริบิวต์
34	display stream	แสดงสตรีมของตัวอักษร
35	display text	แสดงเท็กซ์ของตัวอักษร
36	display stream with attribute	แสดงสตรีมของตัวอักษรพร้อมสตรีมของแอตทริบิวต์
37	display text with attribute	แสดงเท็กซ์ของตัวอักษรพร้อมสตรีมของแอตทริบิวต์

ตารางที่ 1 แสดงหมายเลขฟังก์ชันของ PDD ในส่วนแสดงผลบนจอภาพ

นี้ ฟังก์ชัน PDD เขียนขึ้นด้วยภาษาแอสเซมบลีเพื่อจัดการกับบิตแสดงผลต่าง ๆ บนจอภาพโดยตรง

การเรียกฟังก์ชันเหล่านี้จะเรียกตามหมายเลขฟังก์ชัน โดยแต่ละตัวจะมีการย้ายและรับค่าพารามิเตอร์ตามที่กำหนดของฟังก์ชัน รูปแบบของการเรียกใช้ เป็นดังนี้

vdo-func (funct__no,...)

เช่น

vdo-func(0) เป็นการเซตระบบวิดีโอ

vod-func(2,ON) เป็นการกำหนดโหมดกราฟิกให้ออน

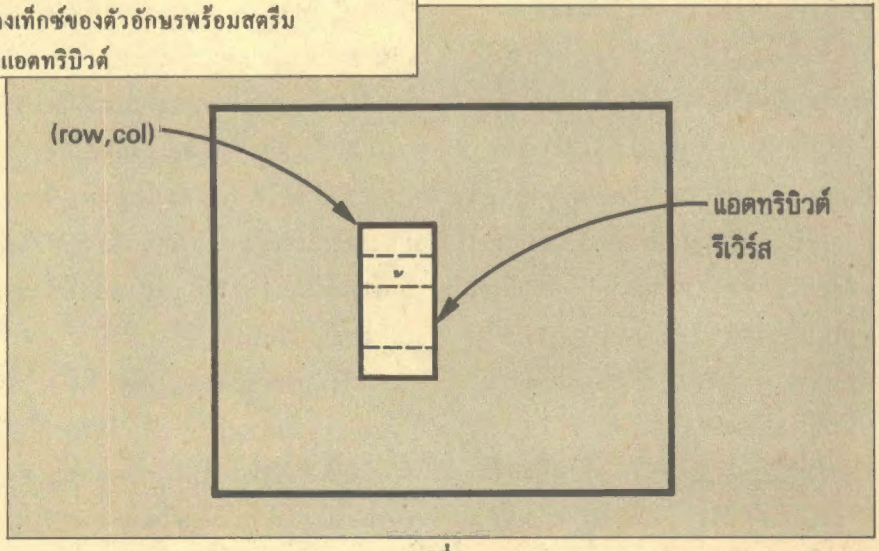
vdo-func(16,' ',col,row, ATTR-REVERSE) เป็นการแสดงไม้โทตรงพิกัดสดมภ์และแถวที่กำหนดด้วยแอตทริบิวต์แบบรีเวิร์ส ดังรูปที่ 2

การแสดงผล ้ นี้ ถ้าตำแหน่งเดิมไม่มีสระอยู่ก็จะลงมาตำแหน่งที่ลดตัวลง แต่ถ้ามีก็จะแสดงตำแหน่งที่สูงขึ้น และถ้าต้องการแสดงผลสระ ี ลงในตำแหน่งพิกัดสดมภ์และแถวเดิม เช่น

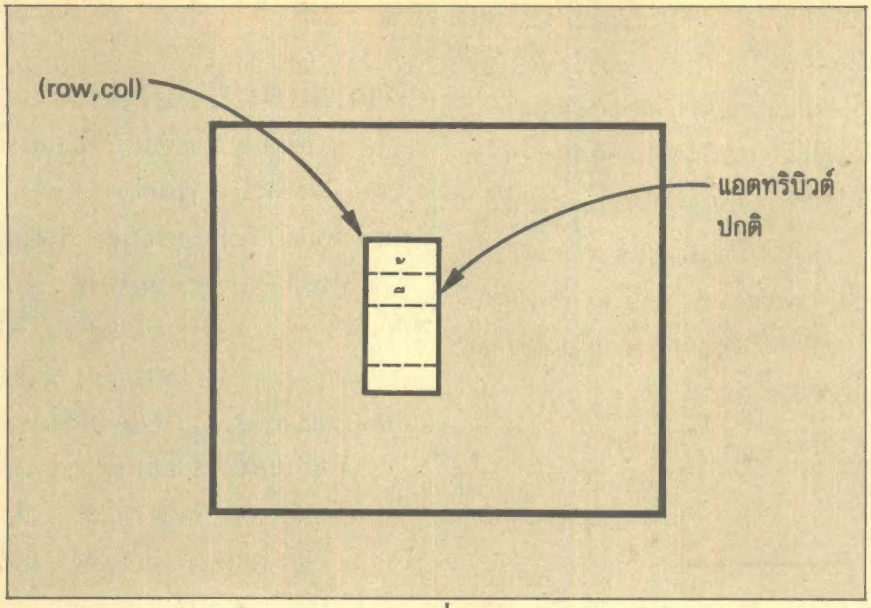
vdo__func(17,'ี',col,row, ATTER__NORMAL)

สระ ี ก็จะไปแสดงยังพิกัดแถวและสดมภ์ในตำแหน่งดังรูปที่ 3

สังเกตจากรูปที่ 2 และรูปที่ 3 จะเห็นว่า รูปที่ 2 ไม้โทจะแสดงในตำแหน่งของสระบน(ในกรณีที่ไม่มีสระบนอยู่) พร้อมแอตทริบิวต์ทั้งหมดจะเป็นแบบรีเวิร์ส รูปที่ 3 ไม้โทจะขยับขึ้นข้างบนโดยที่สระ ี จะแสดงตรงตำแหน่งสระบนแทนพร้อมทั้งเปลี่ยน



รูปที่ 2



รูปที่ 3

แอตทริบิวต์จากเดิมจากรีเวิร์สเป็นแบบปกติ

ตำแหน่งวรรณยุกต์ สระบน และ อุ อู เป็นตำแหน่งที่โปรแกรมในฟังก์ชันการแสดงผลจะจัดการให้อย่างอัตโนมัติ ในกรณีที่ไม่มีการแสดงผลจะแสดงว่างไว้โดยใส่รหัส 160 หรือ A0H แต่ในตำแหน่งพยัญชนะถ้าหากว่างจะเก็บรหัส 32 หรือ 20H

ดังนั้นโครงสร้างของอักษรภาษาไทยจะเป็นดังนี้

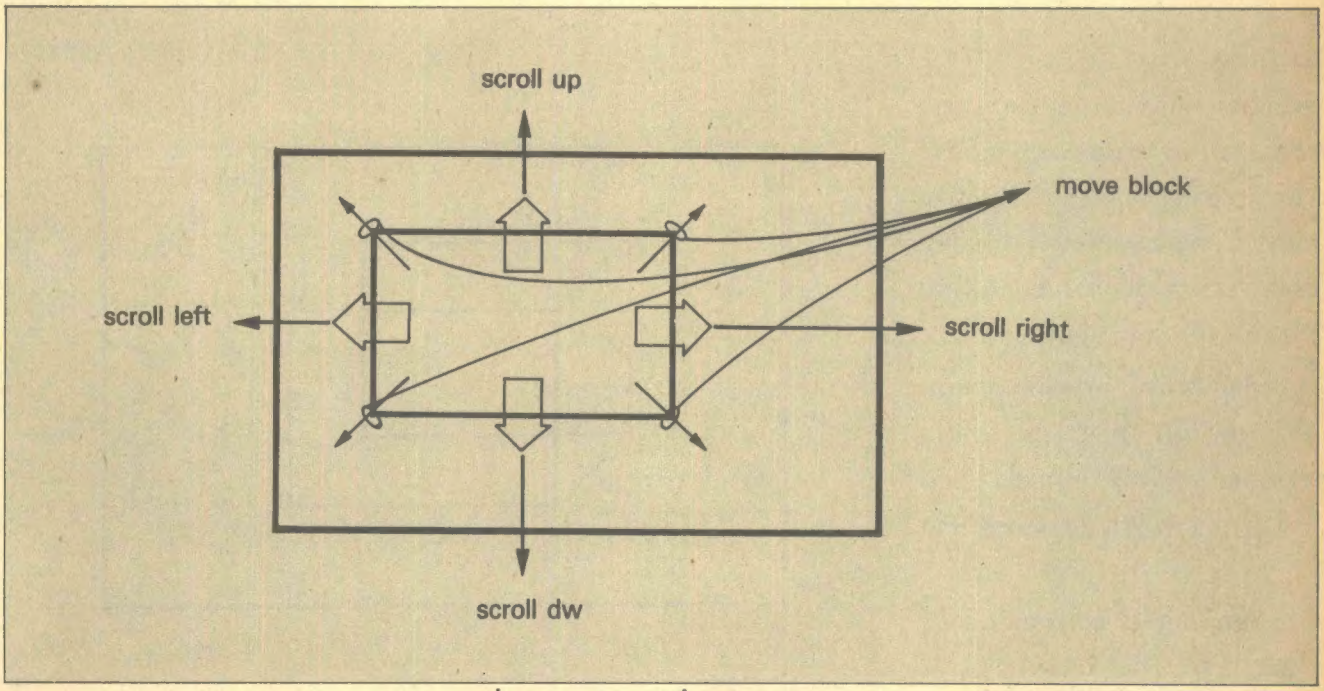
```
struct {
    unsigned char vana, sara,
    pyan,sara__l,attr; a__thai
    __char;
    vdo__func (27,&a_thai__
    char,row,col);
```

ตัวอย่างข้างบนจึงเป็นวิธีการอ่านตัวอักษรภาษาไทยแบบโครงสร้าง ซึ่งจะได้ผลลัพธ์ ดังนี้

```
a__thai__char.vana = '◌ั'
a__thai__char.sara = '◌า'
a__thai__char.pyan = 20H
a__thai__char.sara__l = A0H
a__thai__char.attr = ATTR__
NORMAL
```

โครงสร้างที่อ่านได้นี้จะนำไปใช้ประโยชน์ต่อไปได้ทันที สังเกตว่าการพิจารณาตัวอักษรไทยนั้น เรามองโครงสร้างเป็นแบบ 4 ระดับได้เลย โดย 4 ระดับนี้จะมาใช้ประโยชน์ต่อไปได้

อนึ่งจากตอนก่อนจะพบว่าเรากำหนดให้การเชื่อมต่อกับ TKS ด้วยการมองภาษาไทยออกเป็นสตรีม และเท็กซ์ โดยสตรีมจะหมายถึงการมองโครงสร้างภาษาไทยเรียงกันแบบเชิงเส้น และปิดท้ายข้อความนี้ด้วยรหัส 00 หรือ null code ส่วนเท็กซ์หมายถึงการมองภาษาไทยออกเป็น 4 บรรทัด บรรทัดแรกคือบรรทัดบนสุด บรรทัดที่สองเป็นที่อยู่ของสระบน บรรทัดที่ 3 เป็นที่อยู่ของพยัญชนะ และบรรทัดล่างเป็นที่อยู่ของสระอุอูทุกบรรทัดจะเก็บปิดท้ายด้วยรหัส 00 เช่นกัน ดังนั้นในโหมด



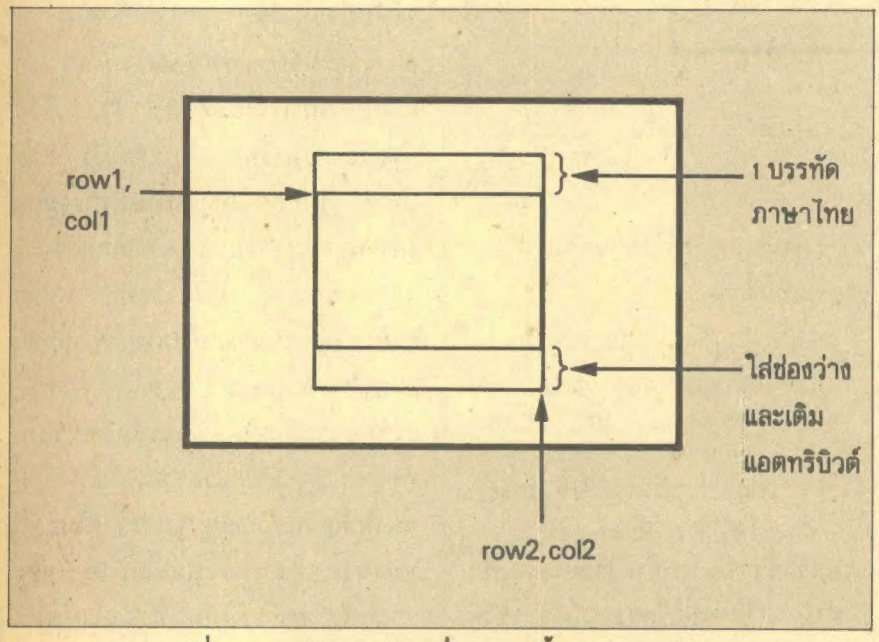
รูปที่ 4 โครงสร้างการเลื่อนภาพใน PDD

เท็กซ์ 4 บรรทัดการแสดงผลจึงเท่ากับภาษาไทย 1 บรรทัด

ฟังก์ชันการเลื่อนภาพ (scroll)

การเลื่อนภาพ (scroll) ของภาษาไทยมีลักษณะพิเศษแตกต่างจากระบบไป โดยกำหนดขนาดของช่องหน้าต่าง คือ col1,row1 และ col2,row2 โดยโดยเลื่อนขึ้นเป็นระยะ 1 บรรทัด โดยมีการใส่แอตทริบิวต์เป็นแบบปกติให้ด้วย การเลื่อนภาพนี้จะให้ผลได้ดังรูปที่ 5

การเลื่อนภาพนี้จะเลื่อนไปทางซ้ายขวาขึ้นบนลงล่างก็ได้ การเรียกใช้นี้สามารถเรียกแบบเคลื่อนย้ายทั้งบล็อกได้เช่นกัน



รูปที่ 5 แสดงตัวอย่างการเลื่อนภาพขึ้นบน 1 บรรทัด

การเคลื่อนย้ายภาพ

การกำหนดการเลื่อนภาพลักษณะ scroll ได้กล่าวถึงแล้ว แต่เรามีฟังก์ชันหมายเลข 11 ที่เลื่อนกรอบหน้าต่างไปตามที่กำหนดได้ รูปแบบของฟังก์ชันเป็นดังนี้

vod-func(11,col1,row1,col2, row2,col3,row3,ATTR__NORMAL)

มีความหมายย้ายช่องหน้าต่างที่กำหนดโดย col1,row1 และ col2, row2 ไปยังตำแหน่ง col3,row3 ผลของภาพจะเป็นดังรูปที่ 6

บริเวณที่แลเงาจะได้รับการแสดงด้วยการกำหนดให้เป็นช่องว่าง และแอตทริบิวต์เป็นแบบปกติ (ATTR__NORMAL)

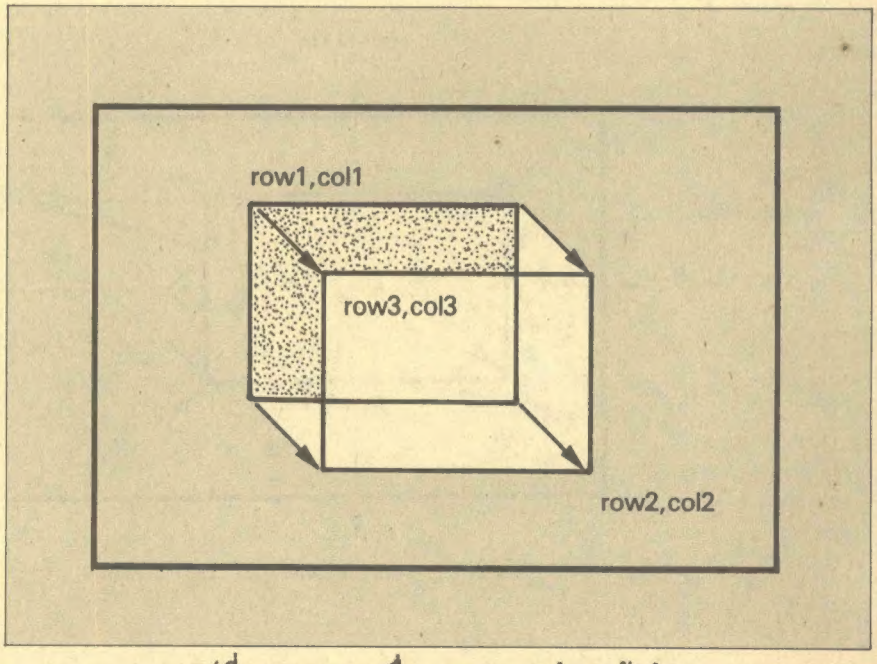
ปกติฟังก์ชันเคลื่อนย้ายภาพ(ฟังก์ชันหมายเลข 11)นี้ จะทำงานช้ากว่าฟังก์ชันการ scroll แบบปกติ คือ up, dw,left,right ดังนั้นถ้าหากต้องการเลื่อนภาพแบบ scroll ให้ใช้ฟังก์ชันเกี่ยวกับ scroll โดยตรงจะดีกว่า

เดิมที่เป็นภาษาอังกฤษ เพราะการแสดงผลในส่วนของภาษาไทยมีส่วนของสระบนล่างเกี่ยวข้องอยู่ด้วย สำหรับใน PDD นี้ได้มองการเลื่อนภาพเป็นแบบช่องหน้าต่างโดยมีรูปแบบของฟังก์ชันที่เกี่ยวข้องดังแสดงในรูปที่ 4

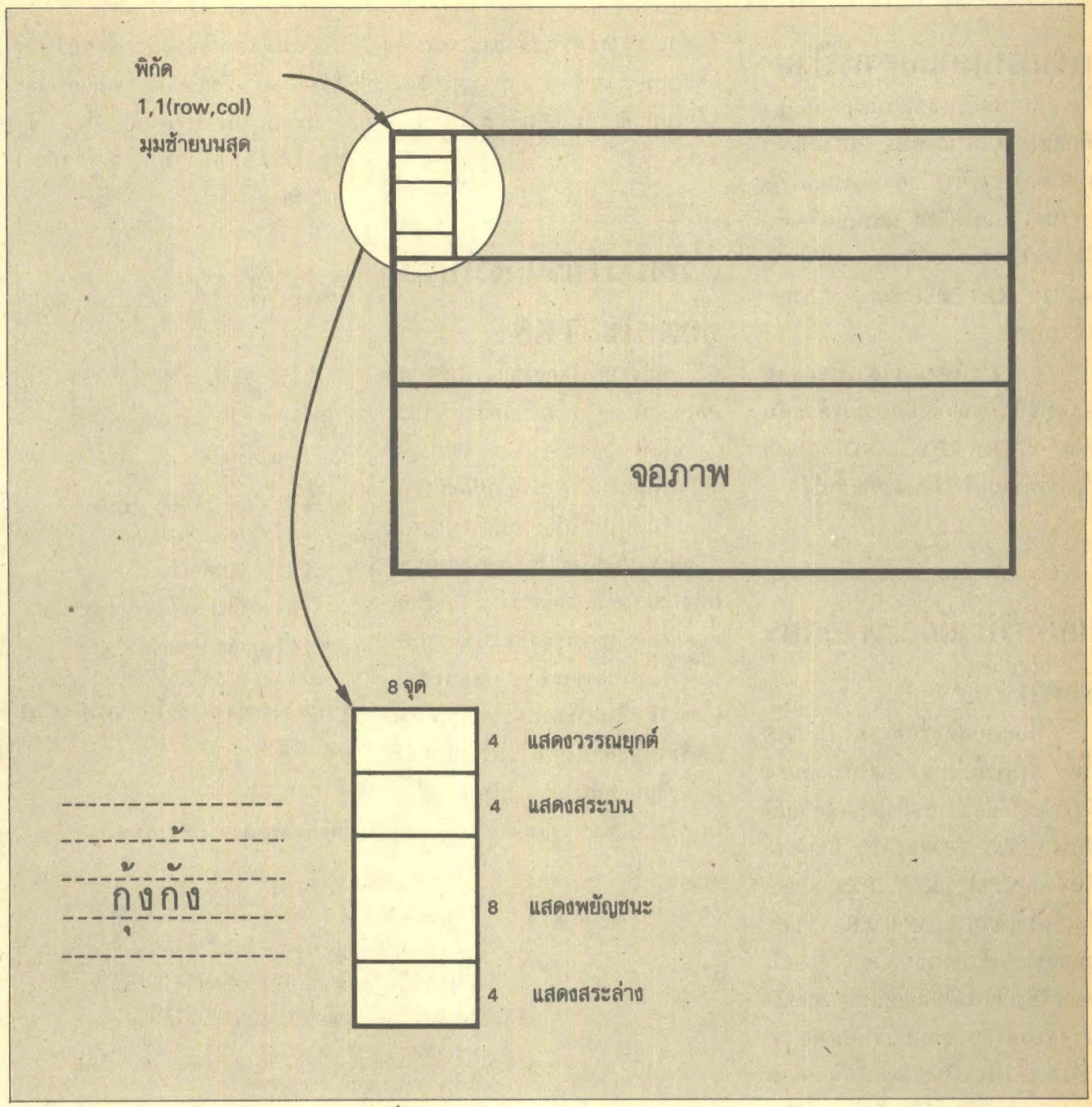
ฟังก์ชันที่ใช้ในการเลื่อนภาพนั้นได้กำหนดไว้ใน PDD โดยกำหนดหมายเลขตั้งแต่ฟังก์ชันหมายเลข 11 ถึง 15 จำนวน 5 ฟังก์ชัน ลองพิจารณาจากฟังก์ชัน 12 ดังนี้

vdo-func(12,col1,row1,col2, row2, 1, ATTR__NORMAL)

ฟังก์ชัน 12 เป็นการเลื่อนภาพขึ้น



รูปที่ 6 แสดงการเลื่อนภาพแบบช่องหน้าต่าง



รูปที่ 7 โครงสร้างการแสดงผลบนจอภาพ

การแสดงผลแบบกราฟิกแต่โปรแกรมประยุกต์เห็นเป็นเท็กซ์

การแสดงผลบนจอภาพใช้หลักการของการแสดงบล็อกของตัวอักษร โดยให้ภาพของตัวอักษรเป็นลักษณะจุด โดยมีความกว้างคอลัมน์ของภาษาไทยที่ใช้ในการแสดงผลแบบปกติกว้าง 8 จุด และสูง 20 จุด โดยได้แบ่งออกเป็น 4 ระดับ ดังแสดงในรูปที่ 7

การแสดงผลบนจอภาพแบบ CGA จะได้จำนวนบรรทัดน้อยลง แต่จะได้จำนวนตัวอักษรต่อบรรทัดเป็น 80 ตัวอักษร ดังนั้นในโหมดที่ต้องการให้แสดงตัวอักษรไทยได้ 25 บรรทัดเต็มต้องใช้จำนวนจุดทางแนวดิ่ง 500 จุด แต่สำหรับจอภาพ VGA สามารถให้จุดได้ถึง 480 จุด ดังนั้นในโหมดแสดงผลของตัวอักษรในจอรุ่นใหม่จะไม่มีปัญหาในเรื่องจำนวนตัวอักษรอีกแล้ว

จากรูปที่ 7 นั้นเงื่อนไขของการแสดงผลเมื่อมีวรรณยุกต์ แต่ไม่มีสระบนจะแสดงวรรณยุกต์ในตำแหน่งสระบนแทน

การกำหนดแอตทริบิวต์

ในการกำหนดลักษณะตัวอักษรที่แสดงผลบนจอภาพใน PDD มีการกำหนดแอตทริบิวต์ในลักษณะแบบปกติธรรมดา แบบรีเวิร์ส และแบบขีดเส้นใต้ โดยใช้ตัวย่อ ATTR_NORMAL, ATTR_REVERSE และ ATTR_UNDERL

อย่างไรก็ตามเราสามารถรวมแอตทริบิวต์ของตัวอักษรเข้าด้วยกัน เช่น ATTR_REV_UNDL หมายถึงการทำแบบรีเวิร์สและขีดเส้นใต้

แอตทริบิวต์และตัวอักษรพิเศษ

ในการแสดงข้อความใน TKS นั้น สามารถกำหนดแอตทริบิวต์ของตัวอักษรได้ถึง 4 แบบด้วยกัน คือ แบบปกติ(ATTR_NORMAL) แบบตรงกันข้าม(ATTR_REVERSE) แบบขีดเส้นใต้(ATTR_UNDERL) และแบบตรงกันข้ามผสมขีดเส้นใต้ แต่มีแอตทริบิวต์พิเศษอีกแบบหนึ่งคือ แอตทริบิวต์ของตัวอักษรแบบกราฟิก(ATTR_GRAPHIC) ใน TKS นั้นนอกจากตัวอักษรภาษาไทยแล้ว ยังมีชุดของตัวอักษรพิเศษ(ตรงกับตัวอักษรพิเศษของไอบีเอ็ม) ซึ่งเอาไว้สำหรับทำกรอบหน้าต่าง รหัสของตัวอักษรพิเศษจะทับกับรหัสภาษาไทย ดังนั้นจะต้องการเซตสวิตช์เปิดตัวอักษรพิเศษนี้ด้วย (ซึ่งสวิตช์ทางซอฟต์แวร์) เมื่อเซตสวิตช์ตัวอักษรพิเศษให้เปิดแล้วจึงจะสามารถแสดงผลตัวอักษรพิเศษนั้นได้ และถ้าต้องการใส่แอตทริบิวต์ในตัวอักษรพิเศษจะต้องนำแอตทริบิวต์กราฟิก (ตามที่กล่าวข้างต้น) มาทำการ OR เข้ากับแอตทริบิวต์ปกติที่ต้องการให้เกิดขึ้นด้วย จากลักษณะจึงสามารถจะทำกรอบหน้าต่างแบบต่าง ๆ บน TKS ได้ตามปกติ (ในตอนที่แล้วในฉบับที่ 45)

ตัวอย่างโปรแกรมประยุกต์ที่ใช้ TKS

หลังจากที่ได้กล่าวถึง TKS มาพอสมควร ต่อไปนี้จะเป็นตัวอย่างโปรแกรมที่เรียกใช้ฟังก์ชันของ TKS ตัวอย่างโปรแกรมนี้ได้แนวความคิดมาจากโปรแกรมที่ทำหน้าที่เป็นบทเรียน(CAI) สำหรับสอนนักเรียน โดยการทำงานของโปรแกรมจะทำการอธิบายถึงตัวบทเรียนที่จะสอน การอธิบายจะปรากฏเป็นข้อความหรืออาจจะมีรูปประกอบด้วย จากนั้นจึงเป็นการถามความเข้าใจของนักเรียนโดยแสดงคำถามพร้อมคำตอบให้นักเรียนเลือกตอบ โปรแกรมที่อยู่ใน TKS นั้นการแสดงอยู่บนโหมดกราฟิกอยู่แล้ว ซึ่งผู้อ่านอาจจะดัดแปลงตัวอย่างโปรแกรมที่ให้สามารถแสดงรูปภาพประกอบบทเรียนได้ด้วย มาดูลักษณะการทำงานโปรแกรมกันก่อน เมื่อเริ่มทำงานโปรแกรมจะรอรับข้อความต่าง ๆ ที่จะใช้ในการแสดง โดยจะปรากฏรูปบนจอภาพดังรูปที่ 8 ซึ่งช่องให้เติมข้อความอยู่ 5 ช่องต่อ 1 บรรทัด

เลขที่	ข้อความ	แถว	หลัก	ต่อไป

F1 = อ่านแฟ้มข้อมูล
F3 = เริ่มทำงาน
F2 = เก็บแฟ้มข้อมูล
F4 = เลิกโปรแกรม

รูปที่ 8 เมื่อทำงานของโปรแกรม CAI (บน TKS)

โดยแต่ละช่องจะมีความหมายดังนี้

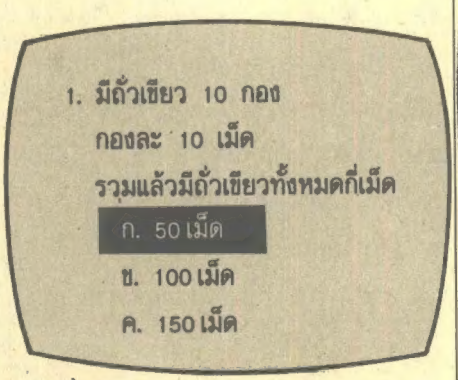
1. ช่องเลขที่ - จะหมายถึงลำดับการแสดงของข้อความ(ในช่องที่ 2) ในกรณีที่ข้อความมีหลายบรรทัดก็ให้ใส่เลขที่เดียวกัน แต่ในกรณีที่ข้อความนั้นใช้สำหรับเป็นตัวเลือกหรือคำตอบของข้อความเลขที่ในบรรทัดบนให้ใส่เครื่องหมายคำถาม "?" แทน
2. ช่องข้อความ - หมายถึงข้อความที่ใช้แสดงผล
3. ช่องแถวและหลัก - จะหมายถึงตำแหน่งแถวและหลักของข้อความในการแสดงผลในจอเฮอร์คิวลิสต์ จะได้ 90 หลักและ 17 แถว ส่วนจอ CGA จะได้ 80 หลักและ 9 แถว
4. ช่องต่อไป - ในช่องนี้จะใช้สำหรับใส่เลขที่ของข้อความที่จะให้ไปทำต่อไป เช่นกรณีที่คำตอบของคำถาม(ในช่องเลขที่จะ ?) เมื่อเลือกคำตอบนี้ ถ้าในช่องนี้มีเลขที่ของข้อความอื่นก็จะไปทำงานในเลขที่นั้น ถ้าใส่ศูนย์ในช่องนี้หมายถึงหยุดการทำงาน

จะขอยกตัวอย่างเพื่อความเข้าใจสักตัวอย่างหนึ่ง ลักษณะเป็นการเติมข้อความซึ่งเป็นข้อสอบ พร้อมคำตอบให้

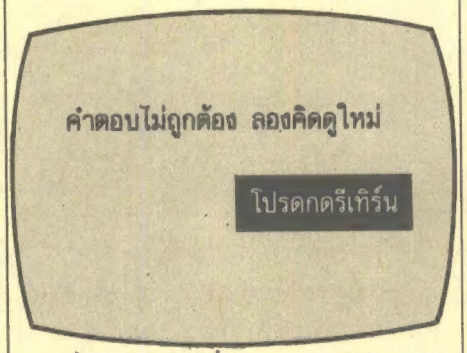
เลขที่	ข้อความ	แถว	หลัก	ต่อไป
1	แบบทดสอบความเข้าใจบทเรียนคณิตศาสตร์	5	10	
1	ยินดีต้อนรับเข้าสู่บทเรียน	7	25	
?	โปรดกดรีเทิร์น	9	60	2
2	1. มีถั่วเขียว 10 กอง กองละ 10 เม็ด รวมแล้ว	2	10	
2	มีถั่วเขียวทั้งหมดกี่เม็ด	3	16	
?	ก. 50 เม็ด	5	25	3
?	ข. 100 เม็ด	7	25	4
?	ค. 150 เม็ด	9	25	3
3	คำตอบไม่ถูกต้อง ลองคิดดูใหม่	3	10	
?	โปรดกดรีเทิร์น	9	60	2
4	คำตอบถูกต้อง เก่งมาก, ทำข้อต่อไป	3	10	
?	โปรดกดรีเทิร์น	9	60	5
5	จบการทดสอบความเข้าใจบทเรียนคณิตศาสตร์	3	20	
?	พบกันใหม่คราวหน้าสวัสดี	7	30	0

รูปที่ 9 เมื่อกดคีย์ F3 โปรแกรมจะแสดงผลตามข้อความที่กำหนดไว้

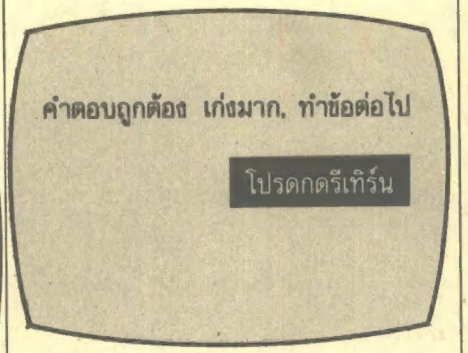
เลือก เมื่อกดคีย์ F3 ให้โปรแกรมแสดงผลตามข้อความที่กำหนดดังรูปที่ 9 และในส่วนที่เป็นคำถามคือ "โปรดกดรีเทิร์น" จะถูกแสดงผลแบบรีเวิร์ส และรอรับการกดคีย์ RETURN ซึ่งถ้ากด RETURN ก็จะปรากฏข้อสอบและคำตอบ ดังในรูปที่ 10 ซึ่งในส่วนของคำตอบซึ่งเป็นการแสดงผลแบบ REVERSE นั้นเราสามารถจะเลือกคำตอบได้โดยกดคีย์ลูกศรเลื่อนเคอร์เซอร์ขึ้นลง แล้วกด RETURN เพื่อเลือกคำตอบนั้น ในกรณีที่เราเลือกคำตอบข้อ ก. หรือข้อ ค. ที่เป็นคำตอบที่ผิดก็จะปรากฏข้อความดังในรูปที่ 11 แล้วกลับมายังรูปเดิมอีกครั้งเมื่อกด RETURN ส่วนในเลื่อนคำตอบข้อ ข. นั้นก็จะปรากฏข้อความดังรูปที่ 12 และเมื่อกด RETURN ก็จะข้ามไปทำข้อต่อไป



รูปที่ 10 แสดงคำถามและคำตอบ



รูปที่ 11 ในกรณีที่เลือกคำตอบผิด หลังจากกดรีเทิร์นก็จะกลับไปยังข้อสอบข้อเดิม



รูปที่ 12 ในกรณีที่เลือกคำตอบถูกต้อง หลังจากกดรีเทิร์นก็จะข้ามไปทำยังข้อถัดไป

โปรแกรมตัวอย่างที่ยกมานี้นั้นคือ โปรแกรม CAI ซึ่งเขียนด้วยภาษาซี(ไมโครซอฟต์ซี) และใช้ฟังก์ชันของ TKS ความเร็วในการทำงานและแสดงผลนั้นอยู่ในระดับดี ตัวโปรแกรมจะอยู่ในรูปที่ 13 ท่านผู้อ่านที่สนใจอยากได้โปรแกรม ก็ติดตามสั่งได้จากคอลัมน์ซอฟต์เมล์ในฉบับนี้

ความหวังของการพัฒนาซอฟต์แวร์ที่มีโครงร่างเป็นมาตรฐาน

ตัวอย่างที่เขียนขึ้นมานี้ใช้ภาษาซีเชื่อมโยงกับ TKS การใช้ภาษาซีก็เนื่องจากเป็นภาษาที่มีรูปแบบของการพัฒนาได้อย่างเต็มตัว ผู้เขียนสามารถใช้เทคนิคที่ทำให้เพิ่มประสิทธิภาพของการทำงานได้อีกมาก นอกจากนี้ความหวังสำหรับการเชื่อมโยงเข้าสู่ระบบอื่น เช่น UNIX หรือคอมพิวเตอร์อื่นทำได้ง่าย งานพัฒนาซอฟต์แวร์ระดับมืออาชีพส่วนใหญ่จึงใช้ภาษาซี

สำหรับตัวอย่างของการวางรูปแบบ TKS นี้เป็นความหวังสำหรับประเทศไทยที่ควรจะสร้างมาตรฐานการเชื่อมต่อของซอฟต์แวร์เหล่านี้ขึ้น เพื่อลดความซ้ำซ้อนของงานที่พัฒนาลง และเป็นมาตรฐานของซอฟต์แวร์ในอนาคต สำหรับโปรแกรมประยุกต์ CAI นี้จึงเป็นตัวอย่างโปรแกรมประยุกต์ที่ทำให้เห็นความสำคัญของ TKS และแนวทางการพัฒนาโปรแกรมประยุกต์ภาษาไทย ในตอนต่อไปจะได้นำเอา TKS ในรายละเอียดอื่นมาเสนอพร้อมทั้งโปรแกรมประยุกต์อีกเช่นกัน